

Amendments to the Claims:

This listing of claims will replace all prior versions, and listings, of claims in the application:

Listing of Claims:

1. (Currently Amended) A method for generating ~~an~~ a software application, comprising the steps of:
receiving a functional description of ~~an~~ the software application; and
automatically partitioning the functional description of the software application into a plurality of modules based on parameterized criteria; and
automatically generating software application code for each module.
2. (Currently Amended) The method of claim 1, wherein the functional description of the software application comprises a flowchart description.
3. (Currently Amended) The method of claim 1, wherein the functional description of the software application comprises a markup description.
4. (Currently Amended) The method of claim 1, wherein the parameterized criteria comprises a measure of software application latency.
5. (Currently Amended) The method of claim 4, wherein automatically partitioning comprises:
partitioning the functional description of the software application into a plurality of different partitions; and
using the parameterized criteria to determine which partition, among the plurality of different partitions, provides a minimal software application latency.

6. (Currently Amended) The method of claim 5, wherein the parameterized criteria comprises a cost function, which is based on transmission and compilation time for different size modules, as the measure of software application latency.

7. (Original) The method of claim 5, wherein the parameterized criteria comprises a probability measure for determining a probability of a given path in a partition being traversed.

8. (Canceled)

9. (Currently Amended) The method of claim 1, further comprising automatically generating a controller that can navigate between the modules of the software application.

10. (Currently Amended) The method of claim 1, wherein the step of receiving ~~comprises~~ comprises:

automatically fetching the functional description of the software application from a persistent storage location; and
performing on-line dynamic remodularization of the software application.

11. (Currently Amended) The method of claim 10, wherein performing ~~on-line~~ on-line dynamic remodularization comprises:

adapting the parameterized criteria according to changes in an environment in which the software application is deployed; and

automatically partitioning the functional description of the software application into a plurality of modules based on the adapted parameterized criteria.

12. (Currently Amended) A program storage device readable by a machine, tangibly embodying a program of instructions executable by the machine to perform method steps for generating an a software application, the method steps comprising:

receiving a functional description of ~~an~~ the software application; and
automatically partitioning the functional description of the software application
into a plurality of modules based on parameterized criteria; and
automatically generating software application code for each module.

13. (Currently Amended) The program storage device of claim 12, wherein the
functional description of the software application comprises a flowchart description.

14. (Currently Amended) The program storage device of claim 12, wherein the
functional description of the software application comprises a markup description.

15. (Currently Amended) The program storage device of claim 12, wherein the
parameterized criteria comprises a measure of software application latency.

16. (Currently Amended) The program storage device of claim 15, wherein the
instructions for automatically partitioning comprise instructions for:

partitioning the functional description of the software application into a plurality of
different partitions; and

using the parameterized criteria to determine which partition, among the plurality
of different partitions, provides a minimal software application latency.

17. (Currently Amended) The program storage device of claim 16, wherein the
parameterized criteria comprises a cost function, which is based on transmission and
compilation time for different size modules, as the measure of software application
latency.

18. (Canceled)

19. (Currently Amended) The program storage device of claim 12, further comprising instructions for automatically generating a controller that can navigate between the modules of the software application.

20. (Currently Amended) The program storage device of claim 12, wherein the instructions for receiving the functional description of the software application comprise instructions for for:

automatically fetching the functional description of the software application from a persistent storage location; and
performing on-line dynamic modularization of the software application.

21. (Currently Amended) The program storage device of claim 20, wherein the instructions for performing on-line dynamic modularization of the software application comprise instructions for:

adapting the parameterized criteria according to changes in an environment in which the software application is deployed; and

automatically partitioning the functional description of the software application into a plurality of modules based on the adapted parameterized criteria.

22. (Currently Amended) A computer-based system for generating an a software application, comprising:

a memory device for storing an application partition module;
a processor in communication with the memory device, the processor operative with the application partition module to automatically partitions partition a functional description of an the software application into a plurality of modules based on parameterized criteria; and
a code generator that automatically generates software application code for each module.

23. (Currently Amended) The ~~tool~~ computer-based system of claim 22, further comprising a user interface for enabling a user to generate ~~[[a]]~~ the functional description of an the software application.

24. (Currently Amended) The ~~tool~~ computer-based system of claim 22, wherein the functional description comprises a flowchart description.

25. (Currently Amended) The ~~tool~~ computer-based system of claim 22, wherein the functional description comprises a markup description.

26. (Currently Amended) The ~~tool~~ computer-based system of claim 22, wherein the parameterized criteria comprises a cost function for resource utilization in a computing environment in which the software application is employed.

27. (Currently Amended) The ~~tool~~ computer-based system of claim 26, wherein the computing environment comprises a network environment.

28. (Currently Amended) The ~~tool~~ computer-based system of claim 22, wherein the parameterized criteria comprises a measure of software application latency.

29. (Currently Amended) The ~~tool~~ computer-based system of claim 28, wherein the parameterized criteria comprises a cost function, which is based on transmission and compilation time for different size software application modules, as the measure of software application latency in a network environment.

30. (Canceled)

31. (Currently Amended) The ~~tool~~ computer-based system of claim ~~[[30]]~~ 22, wherein the code generator automatically generates a controller that can navigate between the modules of the software application.

32. (Currently Amended) A method for generating an a network software application, comprising the steps of:
receiving a functional description of [[a]] the network software application; and
automatically partitioning the functional description of the network software application into a plurality of modules based on parameterized criteria of network latency; and
automatically generating network software application code.

33. (Currently Amended) The method of claim 32, wherein the functional description of the network software application comprises a flowchart description or a markup description.

34. (Currently Amended) The method of claim 32, wherein the network software application comprises a client-server application.

35. (Currently Amended) The method of claim 32, wherein the network software application comprises a Web application.

36. (Currently Amended) The method of claim 32, wherein automatically partitioning comprises:
partitioning the functional description of the network software application into a plurality of different partitions; and
using the parameterized criteria to determine which partition, among the plurality of different partitions, provides a minimal network latency.

37. (Currently Amended) The method of claim 32, wherein the parameterized criteria comprises a cost function, which is based on transmission and compilation time for different size modules, as the a measure of network latency.

38. (Canceled)

39. (Currently Amended) The method of claim [[38]] 32, wherein each module comprises an a network software application view page, and wherein automatically generating network software application code comprises generating markup for each view page and generating server-side code for dispatching the view pages.

40. (Currently Amended) The method of claim 39, further comprising automatically generating a controller that can navigate between the view pages of the network software application.